Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A method in a data processing system for dynamically protecting data from damage during execution of processes within the data processing system, the method comprising:

journaling the data to form journaled data, wherein journaling the data comprises maintaining a ~~previous state of the data for subsequent, optional restore of the data to the previous state~~ system audit trail that contains activities occurring within the data processing system during the execution of the processes within the data processing system;

dynamically determining whether a virus is present in the data processing system after journaling of the data has begun; and

responsive to an identification of the virus, restoring the data using the journaled data.

2. (Original) The method of claim 1 further comprising:

responsive to an absence of an identification of the virus, discarding the journaled data.

3. (Original) The method of claim 1, wherein the determining step comprises:

performing pattern matching.

4. (Currently amended) The method of claim 3, wherein the performing step includes:

comparing a set of actions occurring within the data processing system with a set of known virus patterns.

5. (Currently amended) The method of claim 1, wherein the data that is journaled is located in a storage device external to the data processing system.

6. (Original) The method of claim 1 further comprising:

recording a sequence of actions occurring within the data processing system.

7. (Currently amended) The method claim 1, wherein the data that is journaled is data accessed by a process within the data processing system.

8. (Original) The method of claim 1 further comprising:

responsive to an identification of the virus, blocking access to the data by a process accessing the data.

9. (Original) The method of claim 1 further comprising:

responsive to an identification of the virus, generating an indication halting a process accessing the data.

10. (Previously presented) The method of claim 1, wherein the journaled data is accessed by a single process and maintained until a determination is made that the single process is eliminated as a virus candidate.

11. (Original) The method of claim 1, wherein the journaled data is stored in a protected memory accessible only by the method.

12. (Currently amended) The method of claim 11, wherein the journaled data is stored in a data structure located in a protected memory inaccessible by a process executing within the data processing system.

13. (Currently amended) A method in a data processing system for repairing damage to data, the method comprising:

saving a state of a data object in response to a request to perform a write access to the data object by a process executing on the data processing system;

performing pattern matching of a set of actions taken within the data processing system; and

determining whether an unauthorized intrusion has occurred in response to performing pattern matching and if so, initiating a rollback to return the data object back to its saved state.

14. (Original) The method of claim 13, wherein the performing step comprises:

comparing the set of actions to a pattern from a set of patterns to form a comparison;

determining whether the comparison indicates that the unauthorized intrusion has occurred; and

responsive to an absence of the unauthorized intrusion, repeating the comparing step using another pattern from the set of patterns.

15. (Original) The method of claim 13, wherein the performing step comprises:

matching patterns with the set of actions;

determining whether the unauthorized intrusion has occurred;

if an intrusion is absent, determining whether a time threshold has been reached; and

if an absence of a reaching of the time threshold is present, repeating the matching step using another set of actions.

16. (Original) The method of claim 14, wherein match between the pattern and the set of actions identifies an absence of the unauthorized intrusion.

17. (Original) The method of claim 14, wherein match between the pattern and the set of actions identifies a presence of the unauthorized intrusion.

18. (Original) The method of claim 13, wherein the intrusion is caused by a virus.

19. (Original) The method of claim 13, wherein the intrusion is caused by an authorized user input.

20. (Currently amended) The method of claim 13 further comprising:

saving a state of all data objects within the data processing system, wherein the data objects are dynamically accessed by processes executing within the data processing system.

21. (Original) The method of claim 13, wherein the data object is located in a storage device external to the data processing system.

22. (Currently amended) An intrusion protection system for use in a data processing system comprising:

a sensor filter, wherein the sensor filter receives requests to access data within the data processing system from a process executing within the data processing system;

a pattern matcher, wherein the pattern matcher receives actions initiated by the process, compares the actions to a pattern to form a comparison, determines whether an unauthorized intrusion has occurred, generates a first indication in response to an identification of an absence of an unauthorized intrusion, and generates a second indication to restore the data to a prior state in response to an identification of the unauthorized intrusion; and

a journaler, wherein the journaler journals data in response to accessing of the data by the process executing within the data processing system and restores the data to the prior state in response to the indication by the pattern matcher, wherein the data is journaled until the first indication is generated by the pattern matcher.

23. (Original) The intrusion protection system of claim 22, wherein the intrusion protection system is located within an operating system.

24. (Currently amended) A data processing system comprising:

a bus system;

a communications unit connected to the bus system;

a memory connected to the bus system, wherein the memory includes as set of instructions; and

a processing unit connected to the bus system, wherein the processing unit executes the set of instructions to journal the data to form journaled data, wherein journal the data comprises maintaining a ~~previous state of the data for subsequent, optional restore of the data to the previous state~~ system audit trail that contains activities occurring within the data processing system during the execution of the processes within the data processing system; dynamically determines whether a virus is present in the data processing system after journaling of the data has begun; and restores the data using the journaled data in response to an identification of the virus.

25. (Currently amended) A data processing system comprising:

a bus system;

a communications unit connected to the bus system;

a memory connected to the bus system, wherein the memory includes a set of instructions; and

a processing unit connected to the bus system, wherein the processing unit executes the set of instructions to save a state of a data object in response to a request to perform a write access to the data object by a process executing on the data processing system; perform pattern matching of a set of actions taken within the data processing system; and determine whether an unauthorized intrusion has occurred in response to performing pattern matching and if so, initiate a rollback to return the data object back to its saved state.

26. (Currently amended) A data processing system for dynamically protecting data from damage during execution of processes within the data processing system, the data processing system comprising:

journaling means for journaling the data to form journaled data, wherein the journaling means for journaling the data comprises means for maintaining a ~~previous state of the data for subsequent, optional restore of the data to the previous state~~ system audit trail that contains activities occurring within the data processing system during the execution of the processes within the data processing system;

determining means for dynamically determining whether a virus is present in the data processing system after journaling of the data has begun; and

restoring means, responsive to an identification of the virus, for restoring the data using the journaled data.

27. (Original) The data processing system of claim 26 further comprising:

discarding means, responsive to an absence of an identification of the virus, for discarding the journaled data.

28. (Original) The data processing system of claim 26, wherein the determining means comprises:

means for performing pattern matching.

29. (Currently amended) The data processing system of claim 28, wherein the performing means includes:

means for comparing a set of actions occurring within the data processing system with a set of known virus patterns.

30. (Currently amended) The data processing system of claim 26, wherein the data that is journaled is located in a storage device external to the data processing system.

31. (Original) The data processing system of claim 26 further comprising:

recording means for recording a sequence of actions occurring within the data processing system.

32. (Currently amended) The data processing system claim 26, wherein the data that is journaled is data accessed by a process within the data processing system.

33. (Original) The data processing system of claim 26 further comprising:

blocking means, responsive to an identification of the virus, for blocking access to the data by a process accessing the data.

34. (Original) The data processing system of claim 26 further comprising:

generating means, responsive to an identification of the virus, for generating an indication halting a process accessing the data.

35. (Previously presented) The data processing system of claim 26, wherein the journaled data is accessed by a single process and maintained until a determination is made that the single process is eliminated as a virus candidate.

36. (Original) The data processing system of claim 26, wherein the journaled data is stored in a protected memory accessible only by the method.

37. (Currently amended) The data processing system of claim 36, wherein the journaled data is stored in a data structure located in a protected memory inaccessible by the process executing within the data processing system.

38. (Currently amended) A data processing system for repairing damage to data, the data processing system comprising:

saving means for saving a state of a data object in response to a request to perform a write access to the data object by a process executing on the data processing system;

performing means for performing pattern matching of a set of actions taken within the data processing system; and

determining means for determining whether an unauthorized intrusion has occurred in response to performing pattern matching; and

means for initiating a rollback to return the data object back to its saved state if it is determined that an unauthorized intrusion has occurred.

39. (Original) The data processing system of claim 38, wherein the performing means comprises:

means for comparing the set of actions to a pattern from a set of patterns to form a comparison;

means for determining whether the comparison indicates that the unauthorized intrusion has occurred; and

means, responsive to an absence of the unauthorized intrusion, for repeating the comparing step using another pattern from the set of patterns.

40. (Original) The data processing system of claim 38, wherein the performing means comprises:

means for matching patterns with the set of actions;

means for determining whether the unauthorized intrusion has occurred;

means, if an intrusion is absent, for determining whether a time threshold has been reached; and

means, if an absence of a reaching of the time threshold is present, for repeating the matching step using another set of actions.

41. (Original) The data processing system of claim 39, wherein match between the pattern and the set of actions identifies an absence of the unauthorized intrusion.

42. (Original) The data processing system of claim 39, wherein match between the pattern and the set of actions identifies a presence of the unauthorized intrusion.

43. (Original) The data processing system of claim 38, wherein the intrusion is caused by a virus.

44. (Original) The data processing system of claim 38, wherein the intrusion is caused by an authorized user input.

45. (Currently amended) The data processing system of claim 38 further comprising:

saving means for saving a state of all data objects within the data processing system, wherein the data objects are dynamically accessed by processes executing within the data processing system.

46. (Original) The data processing system of claim 38, wherein the data object is located in a storage device external to the data processing system.

47. (Currently amended) A computer program product in a computer readable medium for dynamically protecting data from damage during execution of processes within the data processing system, the computer program product comprising:

first instructions for journaling the data to form journaled data, wherein journaling the data comprises maintaining a ~~previous state of the data for subsequent, optional restore of the data to the previous state~~ system audit trail that contains activities occurring within the data processing system during the execution of the processes within the data processing system;

second instructions for dynamically determining whether a virus is present in the data processing system after journaling of the data has begun; and

third instructions, responsive to an identification of the virus, for restoring the data using the journaled data.

48. (Original) The computer program product of claim 47 further comprising:

fourth instructions, responsive to an absence of an identification of the virus, for discarding the journaled data.

49. (Original) The computer program product of claim 47, wherein the second instructions comprises:

sub-instructions for performing pattern matching.

50. (Currently amended) The computer program product of claim 47, wherein the sub-instructions for performing includes:

instructions for comparing a set of actions occurring within the data processing system with a set of known virus patterns.

51. (Currently amended) The computer program product of claim 47, wherein the data that is journaled is located in a storage device external to the data processing system.

52. (Original) The computer program product of claim 47 further comprising:

fourth instructions for recording a sequence of actions occurring within the data processing system.

53. (Currently amended) The computer program product claim 47, wherein the data that is journaled is data accessed by a process within the data processing system.

54. (Original) The computer program product of claim 47 further comprising:

fourth instructions, responsive to an identification of the virus, for blocking access to the data by a process accessing the data.

55. (Original) The computer program product of claim 47 further comprising:

fourth instructions, responsive to an identification of the virus, for generating an indication halting a process accessing the data.

56. (Previously presented) The computer program product of claim 47, wherein the journaled data is accessed by a single process and maintained until a determination is made that the single process is eliminated as a virus candidate.

57. (Original) The computer program product of claim 47, wherein the journaled data is stored in a protected memory accessible only by the method.

58. (Currently amended) The computer program product of claim 57, wherein the journaled data is stored in a data structure located in a protected memory inaccessible by the process executing within the data processing system.

59. (Currently amended) A computer program product in a computer readable medium for repairing damage to data, the computer program product comprising:

first instructions for saving a state of a data object in response to a request to perform a write access to the data object by a process executing on the data processing system;

second instructions for performing pattern matching of a set of actions taken within the data processing system; and

third instructions for determining whether an unauthorized intrusion has occurred in response to performing pattern matching; and

fourth instructions for initiating a rollback to return the data object back to its saved state if it is determined that an unauthorized intrusion has occurred.

60. (Original) The computer program product of claim 59, wherein the second instructions comprises:

first sub-instructions for comparing the set of actions to a pattern from a set of patterns to form a comparison;

second sub-instructions for determining whether the comparison indicates that the unauthorized intrusion has occurred; and

third sub-instructions, responsive to an absence of the unauthorized intrusion, for repeating the comparing step using another pattern from the set of patterns.

61. (Original) The computer program product of claim 59, wherein the second instructions comprises:

first sub-instructions for matching patterns with the set of actions;

second sub-instructions for determining whether the unauthorized intrusion has occurred;

third sub-instructions, if an intrusion is absent, for determining whether a time threshold has been reached; and

fourth sub-instructions, if an absence of a reaching of the time threshold is present, for repeating the matching step using another set of actions.

62. (Original) The computer program product of claim 60, wherein match between the pattern and the set of actions identifies an absence of the unauthorized intrusion.

63. (Original) The computer program product of claim 60, wherein match between the pattern and the set of actions identifies a presence of the unauthorized intrusion.

64. (Original) The computer program product of claim 59, wherein the intrusion is caused by a virus.

65. (Original) The computer program product of claim 59, wherein the intrusion is caused by an authorized user input.

66. (Currently amended) The computer program product of claim 59 further comprising:

fourth instructions for saving a state of all data objects within the data processing system, wherein the data objects are dynamically accessed by processes executing within the data processing system.

67. (Original) The computer program product of claim 59, wherein the data object is located in a storage device external to the data processing system.